2003 FAA National Software Conference
XML, Databases and CM for Traceability

VEROCEL

Use of XML, Databases and CM to Automate Traceability

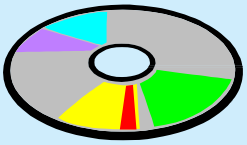
George Romanski
romanski@verocel.com

Paper Based projects

- Prior Projects ('90 – '94) Paper based
  - 1 KLOC ~ 35lb. Paper

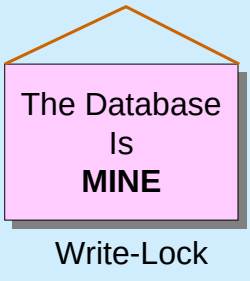
Use of Database – in spreadsheets

- Prior Projects ('90 – '94) Paper based
  - 1 KLOC ~ 35lb. Paper
- Prior Projects ('95 – '99) CD-ROM Delivery
  - Spread-sheets to capture traceability
  - Visual Basic Scripts
  - HTML browsing



Use of Database – Too Simple

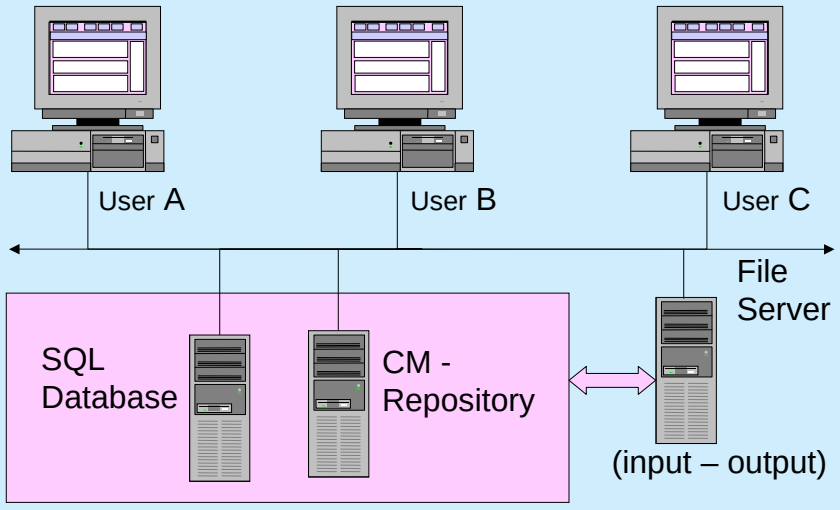
- Prior Projects ('90 – '94) Paper based
  - 1 KLOC ~ 35lb. Paper
- Prior Projects ('95 – '99) CD-ROM Delivery
  - Spread-sheets to capture traceability
  - Visual Basic Scripts
  - HTML browsing
- Start of New Project
  (2000 – for 6 months)
  - Microsoft Access



Evolution to a better way

- Switched to SQL Database for Requirements and Treacability data
- Concurrent read and write to database
- Links between database and CM system
- Process automation/enforcement
- Automated extraction of Life Cycle Data
- Extraction process “qualified” as a tool
- Display of Lifecycle evidence on browseable CD-ROM

Networked Configuration for Life-Cycle Data



Requirements Data

- Requirement Statements in Database
- Tabular requirements may be placed in CM and referenced from database
- Robustness Requirements in Database (trace to robustness tests)
- “Coverage” Requirements in Database (trace to coverage tests)
- Test information may be added by requirement developer
  - For robustness and coverage tests
  - Comments may be added

Requirement Organization

- Hierarchical
- Naming defined for Project
  - AAA.nn.nn.nn-nn
  - Annn.nn.nn.nn-nnn

R12.1		Library Level Requirement
	R12.1.1	Function Level Requirement
	R12.1.2	Function Level Requirement
		R12.1.2-1 Low Level Requirement
		R12.1.2-2
	R12.1.3	
R12.2		
	R12.2.1	

Requirements – Attributes

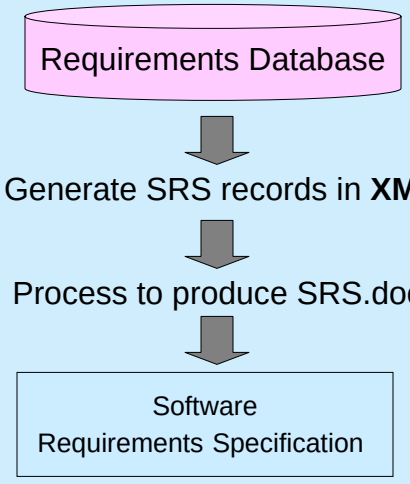
- Requirements_Id (actual, sort_order)

  (R12.1.2, R012.01.02)

- Level (High, Low, Source_File, Function,…)
- Version – of each requirement individually
- State – of each requirement
- Origin (System, Derived, …)
- Verification_Method (Test, Analysis, …)
- Author
- Reviewer

  Auto-check ensures they are different
- Review_Date
- ….

Requirements Specification Document



System Requirements - XML

```
<sysRS>
  <requirement requirementNumber="SYS.1.1">
    <statement>The System shall accept commands
which are supplied using an interactive operator
interface.</statement>
  </requirement>
  <requirement requirementNumber="SYS.1.1.1">
    <statement>The system shall be able to accept at
least one character per second as typed by the
operator.</statement>
    <notes>This time is for the actual characters
typed, and does not take into account the delay
caused while user prompts are output.</notes>
  </requirement>
</sysRS>
```


Auto-generation of documents

SYS.1.1

The System shall accept commands which are supplied using an interactive operator interface.

SYS.1.1.1

The system shall be able to accept at least one character per second as typed by the operator.

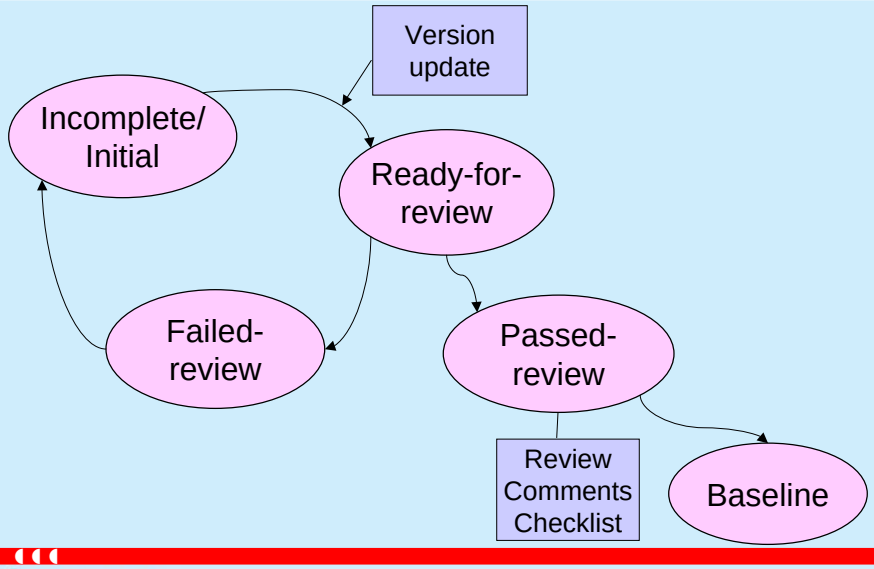
Notes:
This time is for the actual characters typed, and does not take into account the delay caused while user prompts are output.

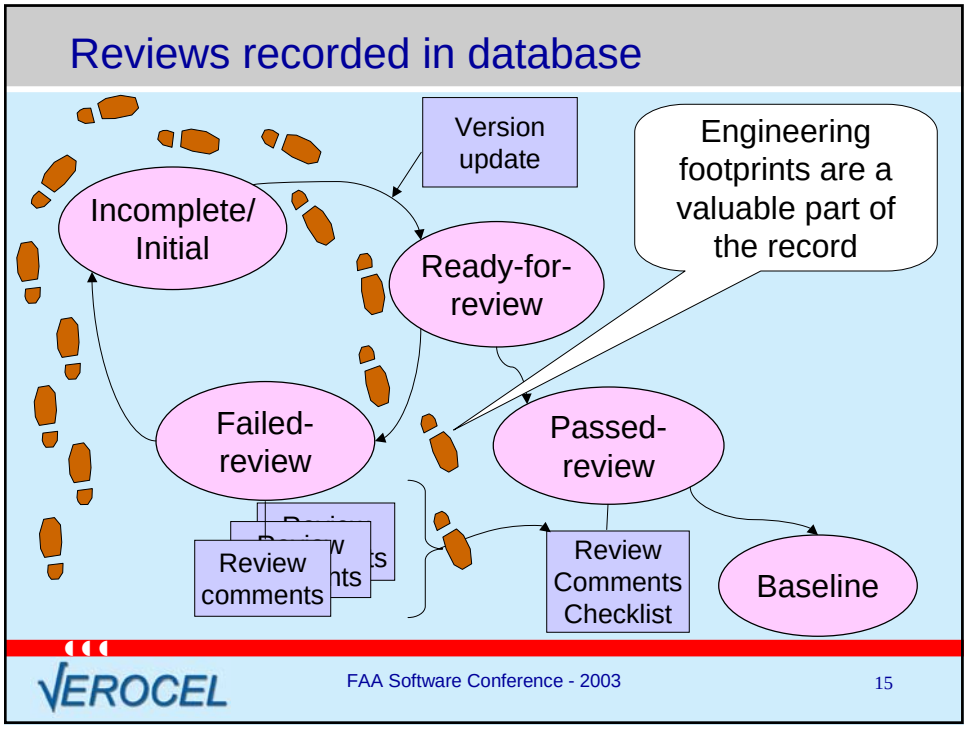
Same for Software Requirements

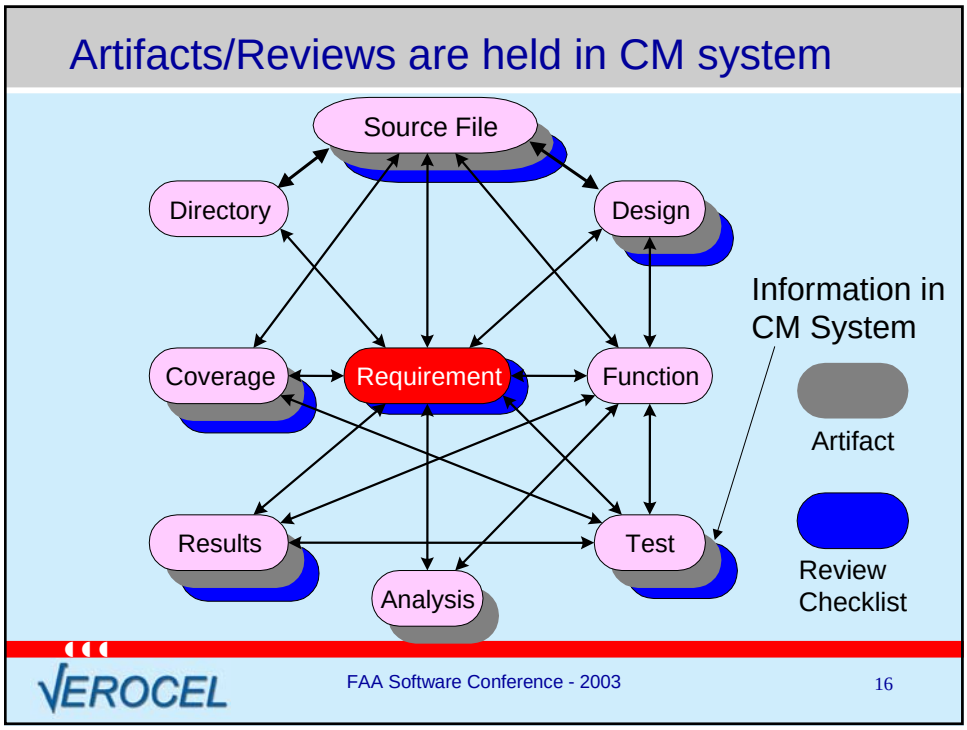
Online vs Document Review

- Review Requirements online
- Review requirements "bottom-up" enforced
- Individual requirement trees can be in different states
- Phase transition tied to requirement
- Partial Requirement documents **can** be auto generated
- Review Requirements in completed Document
- Review requirements "bottom-up" on trust
- Document records complete requirement sets
- Phase transition tied to document

Requirement development states




Reviews recorded in database
Version update
Engineering footprints are a valuable part of the record
Incomplete/ Initial
Ready-for-review
Failed-review
Passed-review
Review comments
Review Comments Checklist
Baseline
VEROCEL
FAA Software Conference - 2003
15

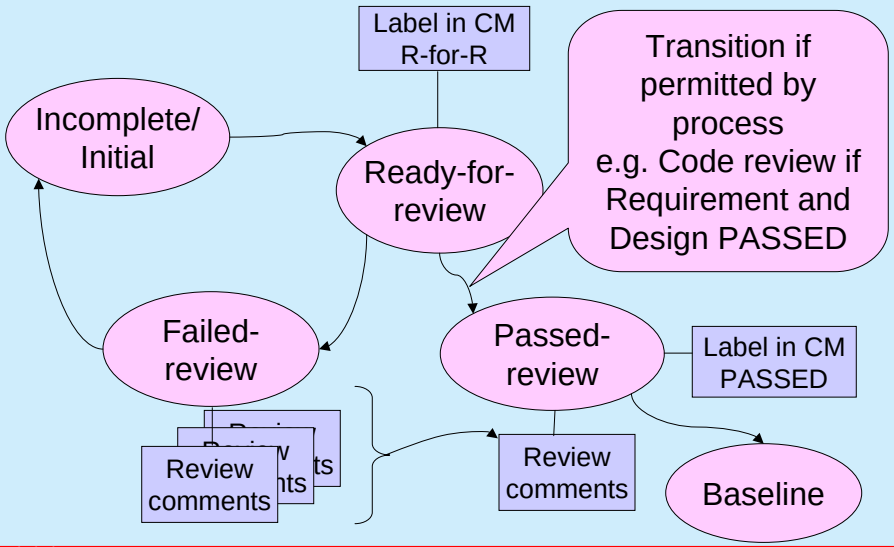

Artifacts/Reviews are held in CM system
Source File
Directory
Design
Coverage
Requirement
Function
Results
Test
Analysis
Information in CM System
Artifact
Review Checklist
VEROCEL
FAA Software Conference - 2003
16

Database to CM references

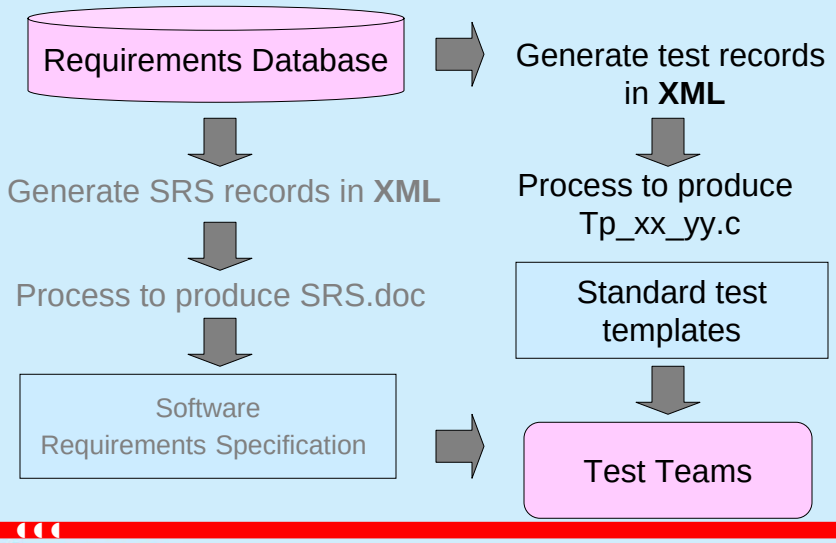
- Life Cycle artifacts always in CM
  - Design Components
  - Code
  - Tests
  - Results
  - Review Checklists
- Artifacts referenced from Database automatically
- Database updates CM records automatically
  - (Passed, Failed, …)

Conjured names by Database
Or user specified names

Same review process for Artifacts



Requirements Database for Test templates

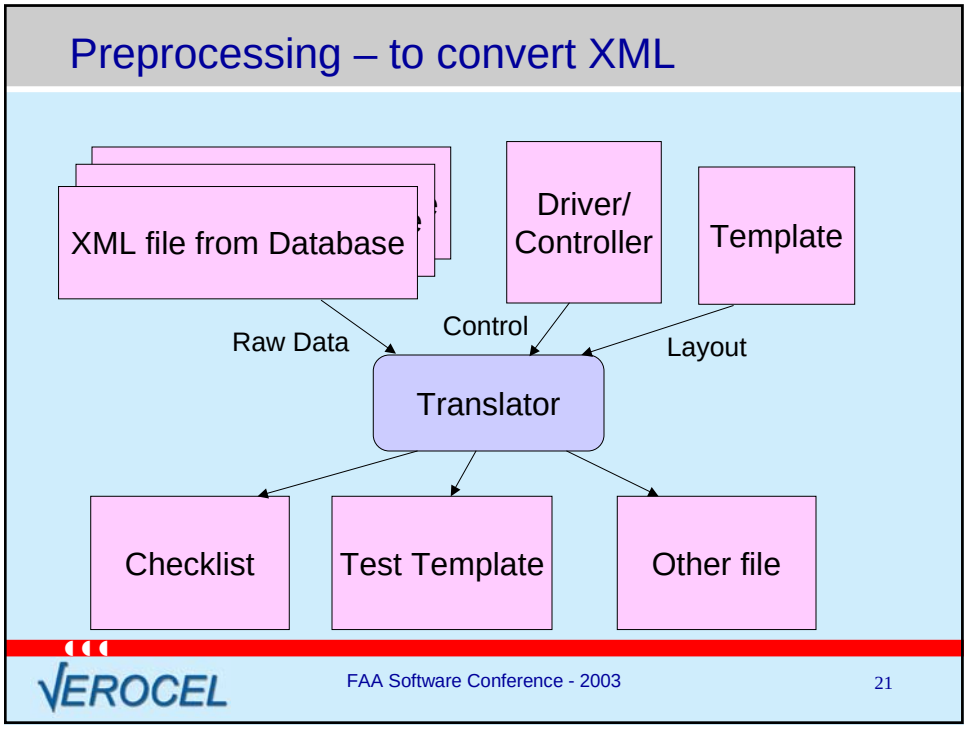


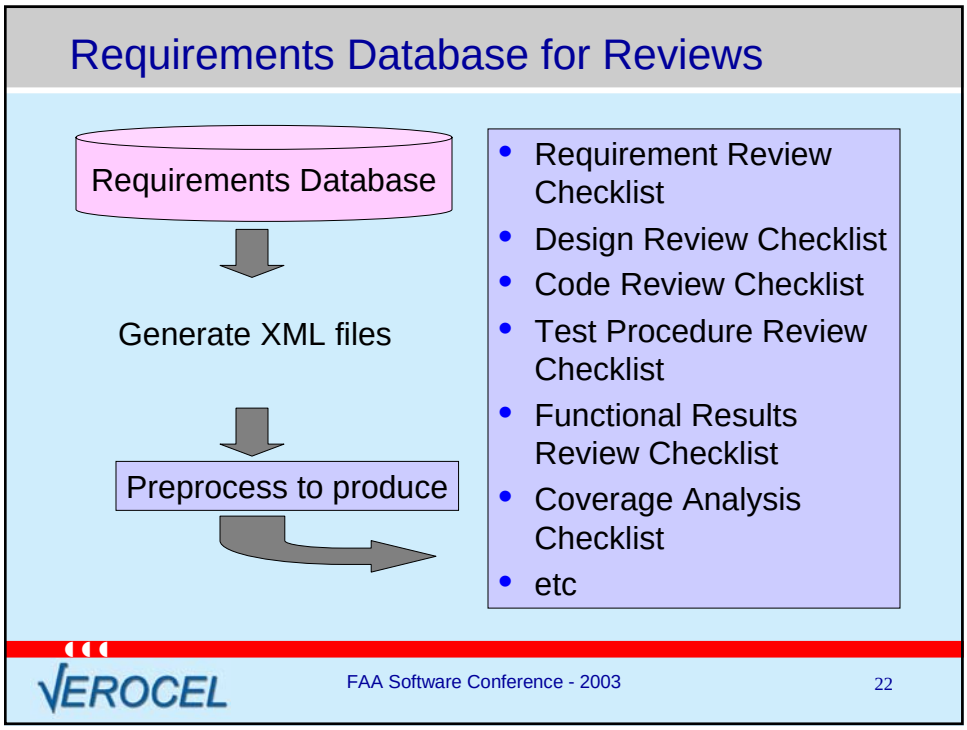
Test File Traceability Information

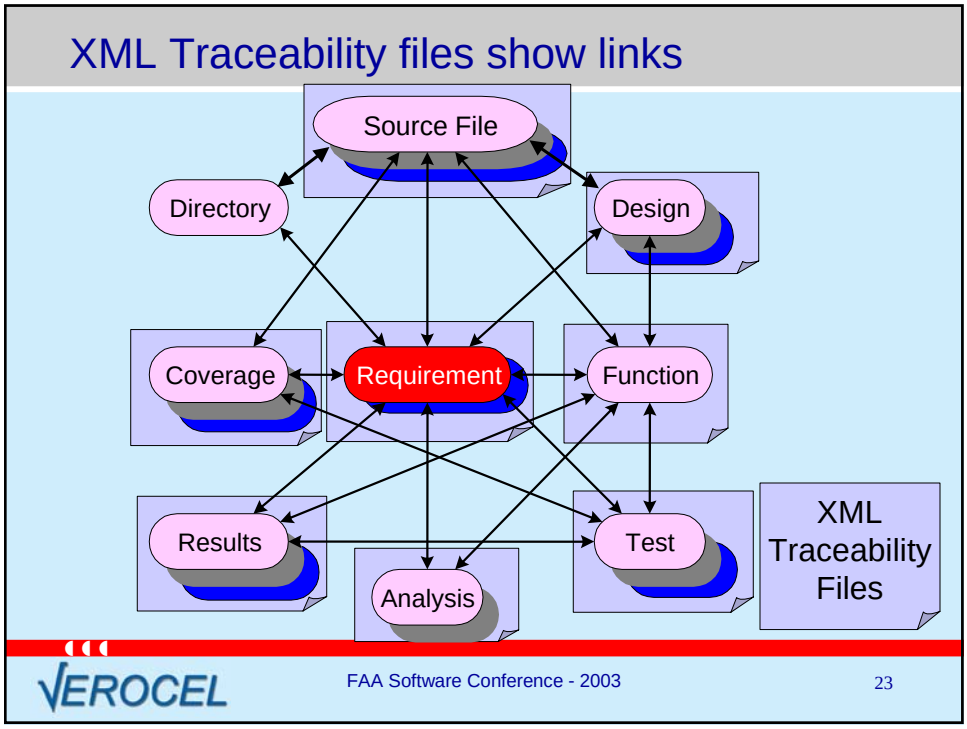
```
/* Traceability Information
   <FILE_NAME>      tp06_01_04__09.c
   <SOURCE_FILE>    tMap.c
   <MODULE>         tMap
   <VISIBILITY>     API
   <HI_REQ>         RR.6.1.4
   <LL_REQ>         RR.6.1.4-9
   <MAX_TEST_CASE>  2
*/
void tp06_01_04__09(void)
{

verHeader ("tp06_01_04__09", “tMap.c",
                  “tMap", "RR.6.1.4-9", 2);
```

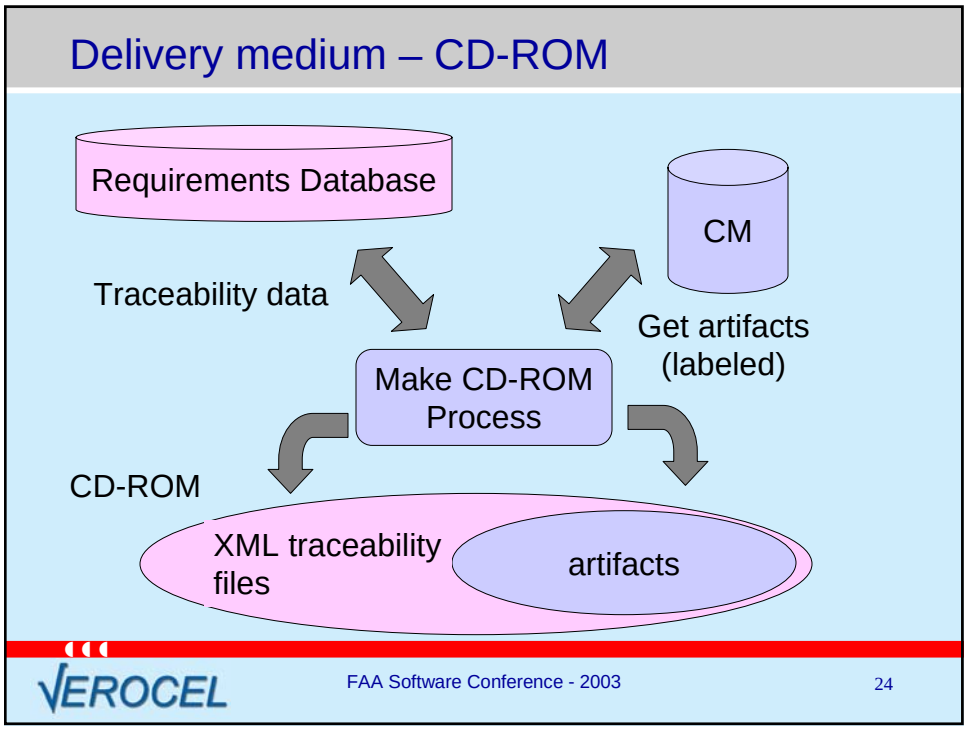
Filled in by Tester

Test Template File is auto generated from XML file

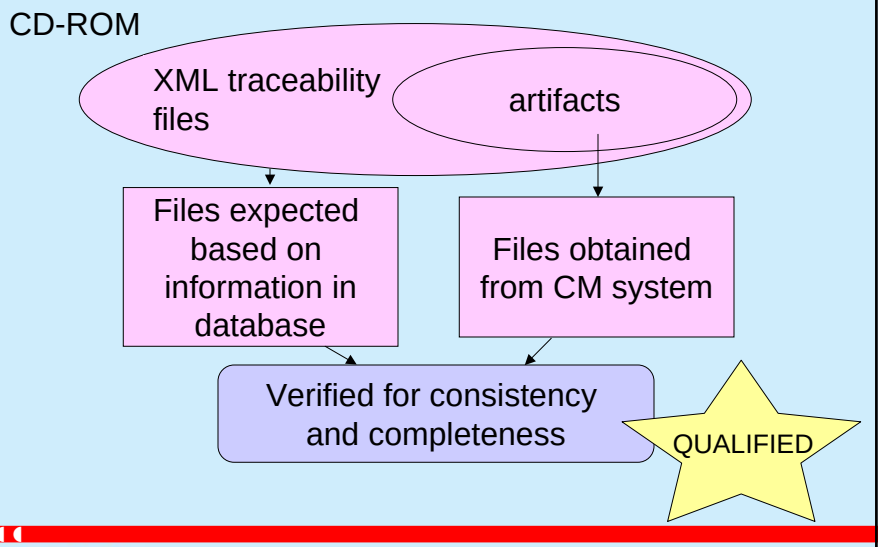

Preprocessing – to convert XML
XML file from Database
Driver/ Controller
Template
Raw Data
Control
Layout
Translator
Checklist
Test Template
Other file
VEROCEL
FAA Software Conference - 2003
21


Requirements Database for Reviews
Requirements Database
Generate XML files
Preprocess to produce
Requirement Review Checklist
Design Review Checklist
Code Review Checklist
Test Procedure Review Checklist
Functional Results Review Checklist
Coverage Analysis Checklist
etc
VEROCEL
FAA Software Conference - 2003
22


XML Traceability files show links
Source File
Directory
Design
Coverage
Requirement
Function
Results
Test
Analysis
XML Traceability Files
VEROCEL
FAA Software Conference - 2003
23


Delivery medium – CD-ROM
Requirements Database
CM
Traceability data
Get artifacts (labeled)
Make CD-ROM Process
CD-ROM
XML traceability files
artifacts
VEROCEL
FAA Software Conference - 2003
24

Verification of Traceability



The Numbers – An RTOS – Level A

Lines of Code	12,000 *
Requirements	1,300
Test Files	720
Lines of test	48,000
Review Files (Checklists)	2,900
XML Traceability Files	14,600

* Includes many support libraries

Use of Tools to manage process

- Requirement captured electronically
- Traceability information added (or conjured by system)
- Requirements may be reviewed online
- Rules may be enforced by tools
  - Reviewer independent from author
  - Low-level requirement reviewed before parent requirement reviewed
  - Verification process order
- Checklists, documents, test templates generated automatically
- Allows parallel development for requirements plus all other artifacts

Experience

- XML easier to manage than HTML
  - Focuses on content rather than layout
- XML based browsers different
  - Auto convert to HTML for presentation
  - Some users STILL want paper (as option)
    - Use HTML to print files
- System requirements imported from Documents
  - Document to database extraction tool (based on SHALLs)
- Fewer mistakes, less re-work
- Parallel Development – much more efficient

CD-ROM based delivery

- Easy to browse for information (compared to paper based)
- Auditors can pre-view on their own machines
- Several reviewers can work in parallel
- Builds confidence

BUT!!!

Take care not to conceal the Processes

Traceability must show HOW as well as WHAT!

e.g. Failed reviews are as valuable as Passed ones